

PR #54194 完整报告

vllm-project/vllm

[Kernel] Make prefix-prefill tiling independent of the KV page size

合并时间: 2026-09-01 09:19

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/54194>

执行摘要

- 一句话: prefill 内核 tile 不再依赖 page 尺寸, 混合模型提速最高 2.78x
- 推荐动作: 值得精读。三个亮点: 一是 vestigial code 的考古方法——从 #31380 引入 PHYSICAL_BLOCK_SIZE 的历史因果中证明 32/32 降级分支已无存在前提, 而不是靠直觉删代码; 二是测试方法论——用 fp32 reference 而非 stock kernel 做对照, 配合 negative control, 能区分“与之前不同”与“错误”, 还补了跨代 LDS 预算验证与 e2e A/B; 三是工程纪律——block-table load 未掩码问题即使有修复方案也因性能代价拆成独立 PR, 不夹带进性能改动。对做 kernel 优化的工程师, 这份 PR 的验证闭环和边界处理方式很值得参考。

功能与动机

PR body 指出: hybrid mamba + full-attention 模型的 attention page 尺寸对齐 mamba page (544、1040、1056...), 是 kernel 对齐的倍数而非 2 的幂, context_attention_fwd 因此把 BLOCK_M/BLOCK_N 从 128/64 降到 32/32。但这个 collapse 是 vestigial 的: 它来自 #31380, 而同一个 commit 引入了 PHYSICAL_BLOCK_SIZE, 此后 _paged_kv_cache_offsets 逐个解析 context token, tile 可以跨 page, 且 BLOCK_M (query 行) 与 BLOCK_N (KV 列) 从不直接索引 paged cache, 因此 tile 尺寸不再依赖 page 是否为 2 的幂。作者还测量出 TRITON_BLOCK_SIZE 的最优值同时依赖 page 尺寸与 head_size (544 = 32x17、1056 = 32x33 时 32 最优, 1040 = 16x65 时 64/128 更优), 这应该交给 autotuner 而不是新增常量, 故本次不扩大改动面。

实现拆解

1. 删除遗留分支 (核心源码改动): 在 vllm/v1/attention/ops/prefix_prefill.py 的 context_attention_fwd 中删除 is_pow2 判断 (real_block_size > 0 and (real_block_size & (real_block_size - 1) == 0)) 及非 2 幂时 BLOCK_M=32 / BLOCK_N=32 的降级分支, 改为无条件 BLOCK_M=128、BLOCK_N=64。grid_fn 仍按 BLOCK_M 对 max_input_len 分块, 调用方 chunked_prefill_paged_decode → RocmAttentionImpl.forward 无需任何改动。
2. 保留 TRITON_BLOCK_SIZE=32: 上下文 tile 宽度刻意不动, 因为扫描显示其最优值依赖 page 尺寸与 head_size, 适合未来的 autotuner 而非新常量; 这也避免了在本次性能 PR 中夹带二次调参。

3. 测试参数化扩展: `tests/kernels/attention/test_prefix_prefill.py` 把 `test_qwen3_nonstandard_block_size` 从单一 shape (64, 1, 128, 544) 扩展为 `NONSTANDARD_BLOCK_SIZE_SHAPES` 三组, 新增 `head_size=256` 且 page 1040/1056 的组合 (覆盖 Qwen3.5-MoE / Qwen3.8-Flash-Next 等模型), 验证大 query tile 下的寄存器和 LDS 压力。
4. 验证体系: 正确性对照从 paged cache gather K/V 构建的 fp32 reference, 而不是对照旧 kernel, 因此能区分“与之前不同”与“错误”; 并用 negative control 验证 harness 本身。随机化 600 组 ragged batch 覆盖 page 128/512/544/1040/1056/2048、`head_size` 64/128/256、MQA/GQA、slidingwindow、non-causal与`k=None`cache-sourced路径。另在 MI355X 与 MI325X 两代 CDNA 上做了 kernel 级与 e2e A/B 验证 (详见评论区精华)。
5. 影响面证明: 通过 import 链证明 `context_attention_fwd` 唯一入口是 `RocmAttentionImpl.forward` → `chunked_prefill_paged_decode`, 且 `ROCM_ATTN` 仅存在于 `vllm/platforms/rocm.py`、`vllm/v1/attention/backends/rocm_attn.py` 与 registry 映射中, CUDA 不会进入该内核; power-of-2 page 原本就走 128/64 分支, 字节级不受影响。

关键文件:

- `vllm/v1/attention/ops/prefix_prefill.py` (模块 预填充内核; 类别 source; 类型 core-logic; 符号 `context_attention_fwd`): 核心变更文件: `context_attention_fwd` 删除基于 page 是否为 2 的幂的 tile 降级分支, 统一 `BLOCK_M=128 / BLOCK_N=64`, 这是全部性能提升的来源。
- `tests/kernels/attention/test_prefix_prefill.py` (模块 内核测试; 类别 test; 类型 test-coverage; 符号 `NONSTANDARD_BLOCK_SIZE_SHAPES`, `test_qwen3_nonstandard_block_size`): 测试配套: 把非标准 page 测试从单一 544/128 shape 扩展为三组参数化 shape, 覆盖 `head_size=256` + page 1040/1056 的大 tile 资源压力场景, 并更新 skip 说明。

关键符号: `context_attention_fwd`, `test_qwen3_nonstandard_block_size`

关键源码片段

`vllm/v1/attention/ops/prefix_prefill.py`

核心变更文件: `context_attention_fwd` 删除基于 page 是否为 2 的幂的 tile 降级分支, 统一 `BLOCK_M=128 / BLOCK_N=64`, 这是全部性能提升的来源。

```
# vllm/v1/attention/ops/prefix_prefill.py — context_attention_fwd 中 tile 选择核心段
real_block_size = v_cache.shape[3]
# 在 #31380 引入 PHYSICAL_BLOCK_SIZE 之前, tile 宽度同时承担 page 寻址职责,
# 所以非 2 的幂 page (544/1040/1056) 必须用 32/32 tile 才能对齐页边界。
# 现在 _paged_kv_cache_offsets 按 PHYSICAL_BLOCK_SIZE 逐个解析 context token,
# tile 可以跨 page; BLOCK_M (query 行) 与 BLOCK_N (KV 列) 也从不索引 paged cache,
# 因此统一使用 128/64 大 tile, 混合模型 prefill 不再被小 tile 拖慢。
BLOCK_M = 128
BLOCK_N = 64
# TRITON_BLOCK_SIZE 是上下文 tile 宽度, 刻意保留 32: 扫描显示最优值同时依赖
# page 尺寸与 head_size (544 = 32x17、1056 = 32x33 时 32 最优, 1040 = 16x65
```

```
# 时 64/128 更优) , 未来应交给 autotuner, 而不是新增常量。
TRITON_BLOCK_SIZE = 32
grid_fn = lambda META: (batch, head, triton.cdiv(max_input_len, META["BLOCK_M"]))
```

tests/kernels/attention/test_prefix_prefill.py

测试配套: 把非标准 page 测试从单一 544/128 shape 扩展为三组参数化 shape, 覆盖 **head_size=256** + page 1040/1056 的大 tile 资源压力场景, 并更新 skip 说明。

```
# tests/kernels/attention/test_prefix_prefill.py — 非标准 page 参数化
# Hybrid mamba + full-attention 模型使用非 2 的幂 attention page (对齐 mamba page) 。
# head_size 256 的 query tile 是寄存器的主要消费者, 用来验证大 tile 的资源压力;
# 1040 与 1056 对应 head_size 256 的 Qwen 混合系列 (如 Qwen3.5-MoE / Qwen3.8-Flash-Next) 。
NONSTANDARD_BLOCK_SIZE_SHAPES = [
    (64, 1, 128, 544),
    (8, 4, 256, 1040),
    (8, 4, 256, 1056),
]
```

```
@pytest.mark.parametrize(
    "num_heads,num_queries_per_kv,head_size,block_size", NONSTANDARD_BLOCK_SIZE_SHAPES
)
@pytest.mark.parametrize("dtype", DTYPES)
@pytest.mark.parametrize("device", CUDA_DEVICES)
@pytest.mark.parametrize("op", OPS)
@torch.inference_mode()
def test_qwen3_nonstandard_block_size(
    num_heads: int,
    num_queries_per_kv: int,
    head_size: int,
    block_size: int,
    dtype: torch.dtype,
    device: str,
    op: Callable,
) -> None:
    """非 2 的幂 page 必须匹配 fp32 reference, 即使 tile 跨 page 边界也一样。"""
    if not current_platform.is_rocm():
        pytest.skip("Non-power-of-2 block sizes are only exercised on ROCm CI.")

    test_contexted_kv_attention(
        num_heads=num_heads,
        num_queries_per_kv=num_queries_per_kv,
        head_size=head_size,
        block_size=block_size,
        sliding_window=0,
        dtype=dtype,
        kv_cache_dtype="auto",
```

```
device=device,  
op=op,  
)
```

评论区精华

review 本身没有内联评论，讨论集中在 issue 评论里作者自述的验证与澄清：

- Fangzhou-Ai 表示 "LGTM can you help to run the CI", shen-shanshan 触发 /ci run, Buildkite CI #86382 通过后批准合并。
- 作者补充了模型级 e2e A/B: Qwen3.5-35B-A3B (TP=1, page 1056) 在 4 x 32768 x 1 配置下从 5.193 s 降到 2.567 s (2.02x), Qwen3.8-2.4T (TP=8, page 1040) 在 4 x 32768 x 1 下为 1.225x, 并解释 "The gain shrinks as the output length grows because decode then dominates the wall clock".
- 作者在 MI325X (CDNA3) 上回应了跨代 LDS 预算疑问: head_dim=256 + BLOCK_M=128 恰好用满 64 KiB LDS, kernel 提速 2.67x (page 544) / 2.00x (page 1056), 最大绝对偏差 6.104e-05。
- 作者用 grep 证据证明 CUDA 不受影响: context_attention_fwd 的唯一 importer 是 chunked_prefill_paged_decode, 其唯一 importer 是 rocm_attn.py, ROCM_ATTN 只存在于 ROCm 平台相关文件中。
- 作者主动披露一个未决边界: 上下文循环的 block-table load 未掩码, 仅当 `query_len >= BLOCK_SIZE` 时安全; 试过 masking 与 clamping 两种修复都损失 5-12% 性能, 决定拆成单独的 PR, 不夹带进本次性能改动。
- block-table load 未掩码的遗留边界 (correctness): 不夹带进本性能 PR, 拆为独立 follow-up; 当前靠 K/V mask 丢弃越界 lanes, 值始终正确。
- MI325X 跨代 LDS 预算验证 (performance): MI325X 验证通过: kernel 提速 2.67x (page 544) / 2.00x (page 1056), 最大绝对偏差 6.104e-05。
- 模型级 e2e 验证补充 (testing): prefill-dominated 配置 1.10x-2.02x 提速; 长输出场景收益稀释符合预期。
- CUDA 是否受影响 (question): NVIDIA 完全不受影响; power-of-2 page 路径原本就走 128/64 分支, 字节级不变。
- no-regression 覆盖 (新增模型对照) (testing): power-of-2 page 模型约 1.00x, 其他 attention 后端不受影响, 无回归。
- CI 触发与合并 (other): Buildkite CI #86382 触发并通过, PR 被批准合并。

风险与影响

- 风险:
 1. bf16 累加顺序变化: 改动使非 2 的幂 page 的输出产生 bf16 噪声内的偏移。作者验证所有 shape 下与 fp32 reference 的最大绝对偏差在改动前后一致, 即新 tiling 不比旧实现离精确算术更远, 但对依赖逐字节稳定输出的用户仍是感知差异。
 2. LDS 预算临界: head_dim=256 + BLOCK_M=128 恰好用满 MI325X 的 64 KiB LDS。作者在 MI355X 与 MI325X 两代 CDNA 上验证通过, 但未来 LDS 更小的设备上存在溢

出风险；且测试保留了 ROCm-only skip, CUDA 侧没有覆盖。

3. 未掩码 block-table load 遗留: context_attention_fwd 上下文循环的 block-table load 未掩码, query_len < BLOCK_SIZE 时地址可能越出行, 当前靠 K/V mask 丢弃 lanes 保证值正确, 但仍是内存边界隐患, 等待 follow-up PR。
4. 既有测试失败基线: MI355X 全文件 8 failed / 184 passed / 224 skipped, 8 个失败集中在 chunked_prefill_paged_decode 的 head_size=128 / num_queries_per_kv=1, 与本改动无关但说明该测试文件已有失败基线, 后续回归判断需要留意。
5. 收益的 shape 依赖性: page 1040 (=16x65) 下上下文 tile 仍为 32, 收益相对较小; decode 主导的长输出场景 e2e 提速被稀释到约 1.05x, 整体收益取决于 prefill 占比。
- 影响: 影响范围集中在 ROCm 上的 hybrid mamba + full-attention 模型 (Qwen3-Next-80B、Qwen3.5-MoE、Qwen3.8-Flash-Next 等): kernel 级 prefill 提速 1.5x-2.78x, 端到端 prefill-dominated 配置 1.05x-2.02x。CUDA 用户与 power-of-2 page 模型 (Llama、Qwen3-8B 等) 无感知变化。对 vLLM kernel 团队而言, 本 PR 确立了一个设计原则: tile 尺寸应与物理 page 解耦; 同时 TRITON_BLOCK_SIZE 的 autotuner 化和 block-table load 掩码修复成为可预期的后续工作项。由于改动只有 2 个文件、23 行净变更, 且 CI 通过, 对团队日常开发影响很小。 - 风险标记: bf16 累加顺序变化, LDS 预算临界, block-table 未掩码读取遗留, 仅 ROCm 验证

关联脉络

- PR #31380 (标题未在本 PR 上下文中提供) 引入 PHYSICAL_BLOCK_SIZE 的 PR: PR body 明确指出 32/32 collapse 来自 #31380, 而该 PR 同时引入 PHYSICAL_BLOCK_SIZE 使 page 寻址与 tile 宽度解耦, 是本改动的因果前提。
- PR #53383 (标题未在本 PR 上下文中提供) 为 spec-decode verify 新增 prefix_prefill.py 独立路径的 PR: PR body 说明它也编辑 prefix_prefill.py, 但走短查询 / 长上下文的独立路由且显式保留 _fwd_kernel 不动; 与本次 tiling 改动文本可能重叠但互相独立。
- PR #38454 (标题未在本 PR 上下文中提供) prefix_prefill 相关 test-only PR: PR body 说明 #38454 为 test-only, 与本 PR 的测试扩展在同一文件区域, 合并时需留意文本重叠。
- PR #53147 [Kernel][Gemma4] Prune Triton sliding-window tiles for multimodal prefixes: 同一功能线 (v1 attention prefill 性能优化): 通过 Triton attention helper 中的 tile 剪枝使 Gemma4 prefill 提速 4 倍, 与本 PR 一样在 tile/ 调度层面压低 prefill 延迟。